What is Claimed is:

1. A device for filtering added information comprising:

a memory for storing table IDs and version numbers of sections for each of the table IDs, a combination of at least one of the sections forming a table which added information defines;

5   a comparing unit for determining match of a table ID included in the section and the table IDs stored in the memory upon reception of the section, and comparing the version number of the matched table ID to the version number of a received section number; and,

a section processing unit for receiving, and processing the present section if it is determined that the version number stored in the table ID the same with the table ID of the

10   section received from the comparing unit is not the same with the version number of the received section.

2. A device as claimed in claim 1, wherein the section processing unit skips the section received at the present time if it is determined at the comparing unit that no table IDs are in match, or the version number stored in the matched table ID is the same with the version number

15   of the received section.

3. A device as claimed in claim 1, wherein the section processing unit automatically sets up a version value stored in the memory with a new version number when a section with a new version is received.

4. A device as claimed in claim 1, wherein, if it is determined that the table ID of the

20   section received from the comparing unit presently is in match with the table ID stored in the

14

memory, the section processing unit determines completion of the table of the version processed presently, and mask enables the version number of the section if it is determined that the table is completed.

5. A device as claimed in claim 4, wherein the section processing unit skips the section received at the present time if the version number of the section received at the present time is mask enabled.

6. A method for filtering added information for receiving only required sections using a memory for storing table IDs and a memory for storing version numbers of the sections separate for each of the table IDs, comprising the steps of:

(1) upon reception of the section, determining match of the table ID included in the section and the table IDs stored in the memory;

(2) if it is determined in the step (1) that there are the table IDs matched, determining the version number included in the received section of being the same with the version number stored in a version memory of the table ID; and,

(3) if it is determined in the step (2) that the two version numbers are not the same, receiving and processing the present section.

7. A method as claimed in claim 6, further comprising the step of skipping the section received at the present time if it is determined in the step (1) that there are no table IDs matched, or if it is determined in the step (2) that the two version numbers are the same.

8. A method as claimed in claim 7, wherein the step of skipping the section is caused when the version number of the section received at the present time is mask enabled.

9. A method as claimed in claim 6, further comprising the steps of;

(6-1) determining completion of the table of the version processed at the present time, if it is determined in the step (1) that the table ID of the section received at the present time and the any one of table IDs stored in the memory are matched, and

(6-1) mask enabling the version number of the section if it is determined in the (6-1) step that the table is completed.

10. A method as claimed in claim 6, wherein the step (1) further includes the step of determining starting of the received section with reference to a pointer field if a payload_syntax _indicator is '1' in a transport packet before the step of determining match of the table ID included in the section and the table IDs stored in the memory.

11. A method as claimed in claim 6, wherein the step (1) further includes the step of determining starting of another section if the next data is not 0×FF after the section comes to an end before the step of determining match of the table ID included in the section and the table IDs stored in the memory.

12. A method as claimed in claim 6, wherein the step (3) further includes the step of automatically updating the version value stored in the version memory with a new version number upon reception of a section having the new version.

16